The operating system used is windows. IDE=PyCharm. Python version 3.7

The library used for target generation is numpy.

The library used for sha256 is hashlib.

The library used for time calculation is time.

The target generation.

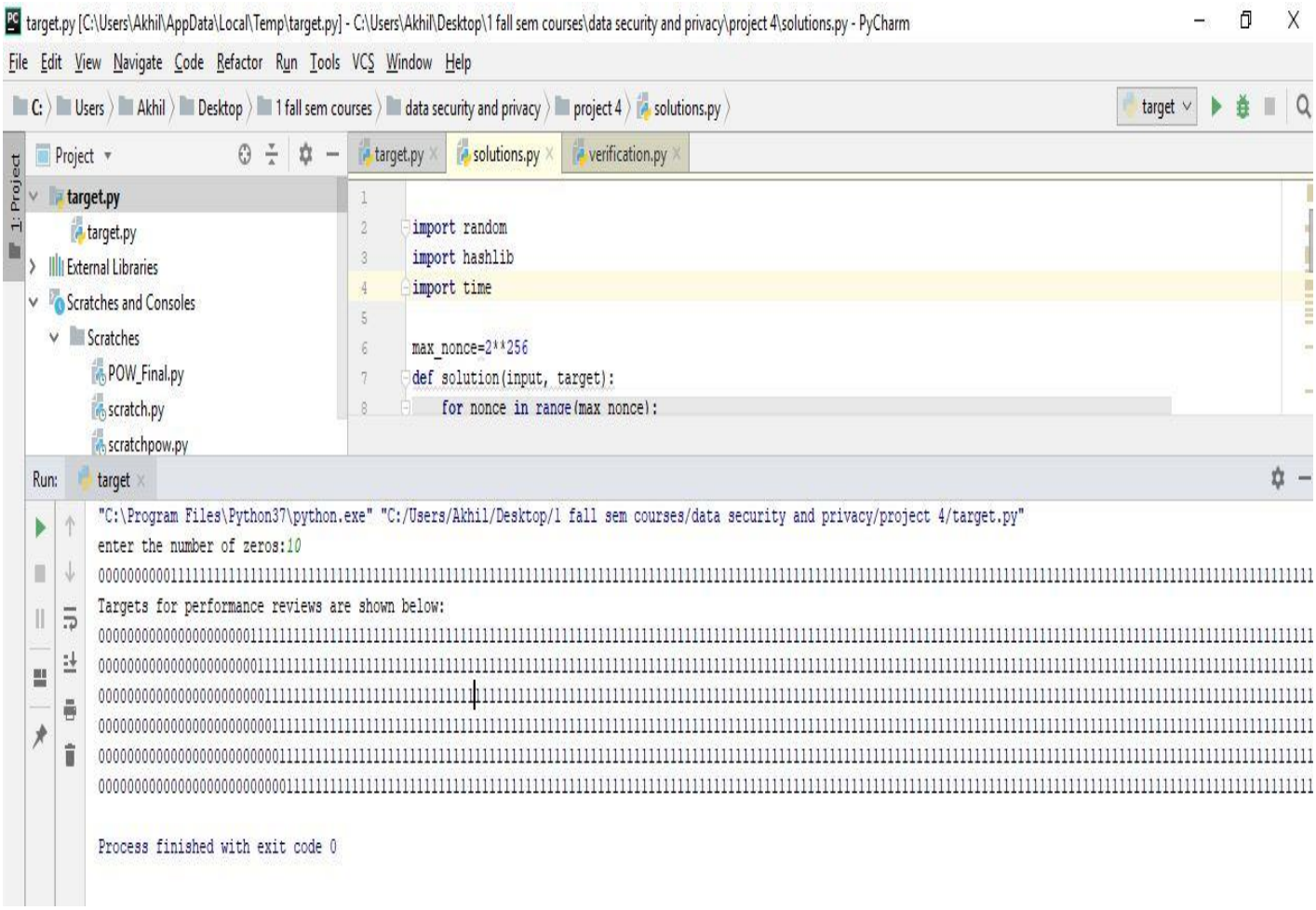
Difficulty is the number of zeros present in the first of a binary number.

For target generation the difficulty used at first is d=10

The code used is:

```
import numpy as py
def main():
    no = input("enter the number of zeros:") # to enter no of zeros
    d=int(no)
    first = py.zeros(d, int)
    last = py.ones(256 - d, int)
    f = "".join(map(str, first))
    l = "".join(map(str, last))
    target = f + l              #concatenation ot two strings
    print(target)
    w1 = open("C:\\Users\\Akhil\\Desktop\\1 fall sem courses\\data security and privacy\\project 4\\target.txt" , "w+") #wr
    w1.write(target)
    w1.close()

def main2():
    for no in range (21,27):            #to display targets with d from 21 to 26
        d=int(no)
        first = py.zeros(d, int)
        last = py.ones(256 - d, int)
        f = "".join(map(str, first))
        l = "".join(map(str, last))
        target2 = f + l
        print(target2)
        no+=1
        w2 = open("C:\\Users\\Akhil\\Desktop\\1 fall sem courses\\data security and privacy\\project 4\\targetforperformanc
        w2.write(target2)
        w2.close()


if __name__ == "__main__":
    main()
    print("Targets for performance reviews are shown below:")
    main2()
```

The output of d=10 and also d=21,22,23,24,25,26 for performance power review is shown below:



These targets with d=10 is written to file target.txt

The targets wit d=21 to 26 is written to file targetforperformance.txt.

Finding Solutions:

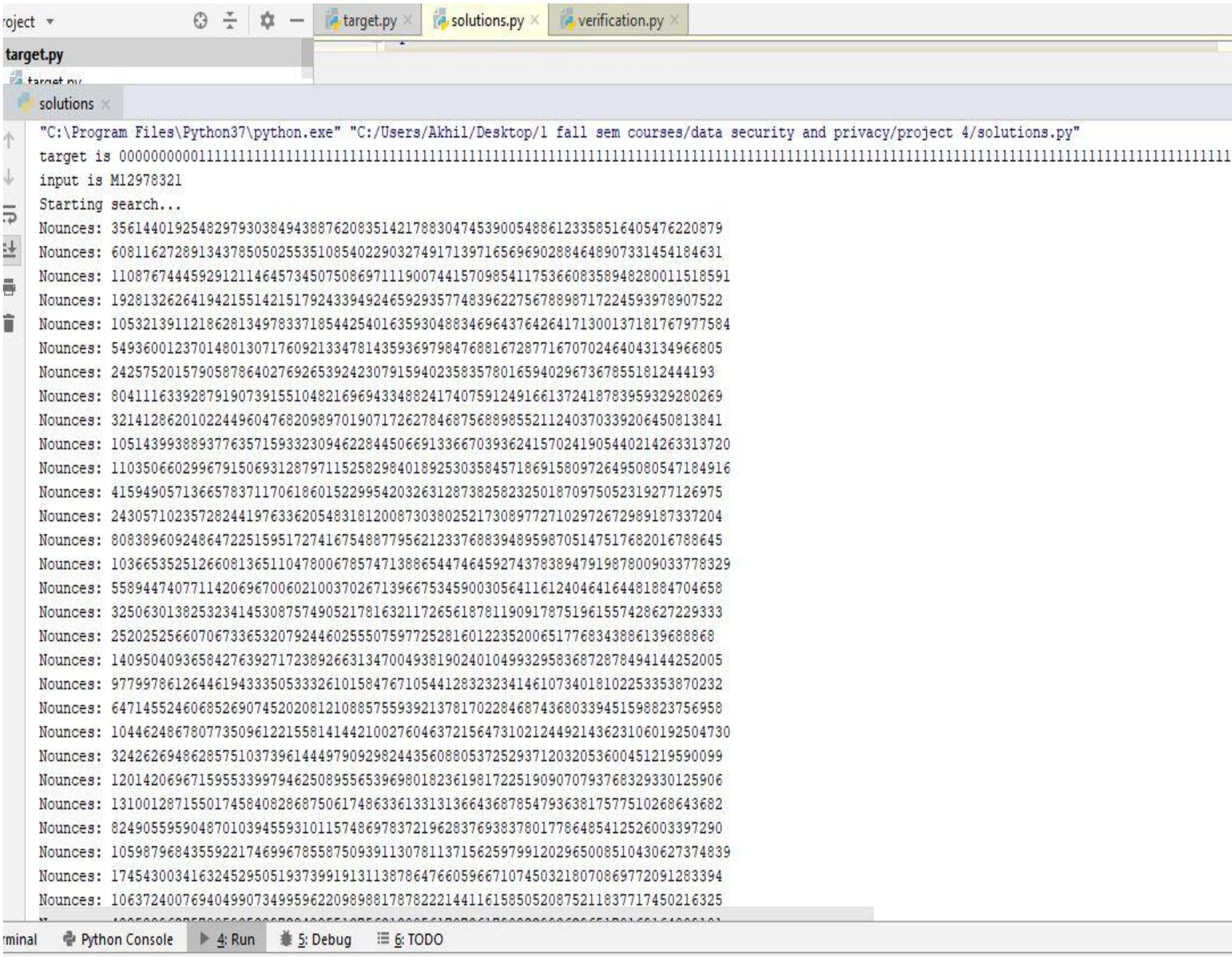
The code used is :

```
max_nonce=2**256
def solution(input, target):
    for nonce in range(max_nonce):
        # to create random numbers of 256bit
        i = random.randint(0, pow(2, 256))
        hash_result = hashlib.sha256((str(input) + str(i)).encode('utf-8')) #SHA256encryption
        hex_dig = hash_result.hexdigest()                                   #converting to hexadecimal
        hash_bin = str(bin(int(hex_dig,16)))                                #converting to binary
        hash_bin=hash_bin[2:]
        hash_10 = int(hash_bin,2)                                           #converting nounce to decimal
        target_10 = int(target,2)                                           #converting target to decimal
        print ('Nounces:',hash_10)
        if hash_10 <= target_10:
            print(" ")
            print('The final hash value or solution is:',i)                 #solution
            return str(i)
        else:
            i+=1
        if i<= pow(2,256):
            i = 0
```

The code for finding time is:

```
# checkpoint the current time
start_time = time.time()

f = solution(input_in, target_in )
wl = open("C:\\Users\\Akhil\\Desktop\\1 fall sem courses\\data security and privacy\\project 4\\solution.txt", "w+")
wl.write(f)
wl.close()

print(" ")
print("Execution completed")

# checkpoint how long it took to find a result
end_time = time.time()

elapsed_time = end_time - start_time
print("Elapsed Time: %.4f seconds" % elapsed_time)
```

The solution obtained, or output is:



```
Nounces: 53147078150233032586330354821295307316287203674183785010091526970596544349579
Nounces: 30260915972776288663662227637447049452379649635348561136966368331760564188288
Nounces: 50468570383694941594119522062879536832574441414206543122491400722870501959487
Nounces: 25485945063437910081781413077459557495457713997826959081663835208127620497283
Nounces: 36005643695977435910446733375793317861812519300333192018879779220549526342267
Nounces: 14944724994979583880039974509674182605200122403246511683299882243857911252391
Nounces: 75368731744893963064883879255188709544528560197393229265889500567258880554623
Nounces: 50723922604513829077833583700510841702605230734231747671950356141722673201463
Nounces: 32285370341618058772280465229408057691756990524901294964902779723615133254 6
Nounces: 44698148181345291189078308964147030250942429387907242881387017113306394272981
Nounces: 92647691606936263294469158140336699749455919050888848010345208366219116630178
Nounces: 11062834520756314628396737403646075122220531193339486933796572296955799283484 0
Nounces: 98224691703466212731026887298815969946519698229066787384594549319894352277460
Nounces: 12410453815576626208418272072880083479746711543634601338120692835594504636106
Nounces: 10330035284793931162477169944757134162953785566966233424444872334263458662887 7
Nounces: 91647806424377350060886227233521765190558608618623378794978370234470814719509
Nounces: 29807553637171027816087669324193682958813826633160369744201495334138043465319
Nounces: 80896938019599441776753848993194285116071383909112644296074308930170053030978
Nounces: 36308331870473782108587436719792604909890046874129447643071043599336652809642
Nounces: 67452170077611961568945864863292523864538949694923083648675679328195018939 91
Nounces: 78874790230091042364377386489986285015946181466899118621945622845662905243940
Nounces: 39677511340723043688022336015927434174113547033142614969568794509072699631680
Nounces: 39790010787018462702065914301183633709952794978920675723291084637885617748030
Nounces: 11775438352822922515872677196330052905700675196172452471735468741751421191

The final hash value or solution is: 14687154087361411489058732443655953023067053047311230785430994332882182400134

Execution completed
Elapsed Time: 0.0360 seconds

Process finished with exit code 0
```

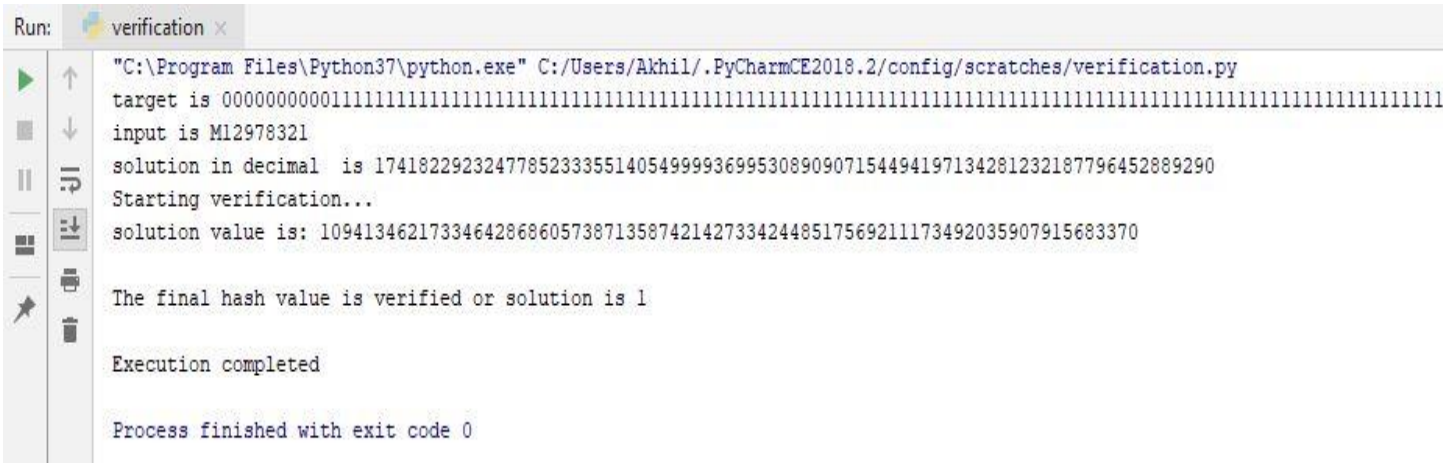
The time taken for d=10 is 0.0360 seconds.

The verification of solution is done:

The code used is:

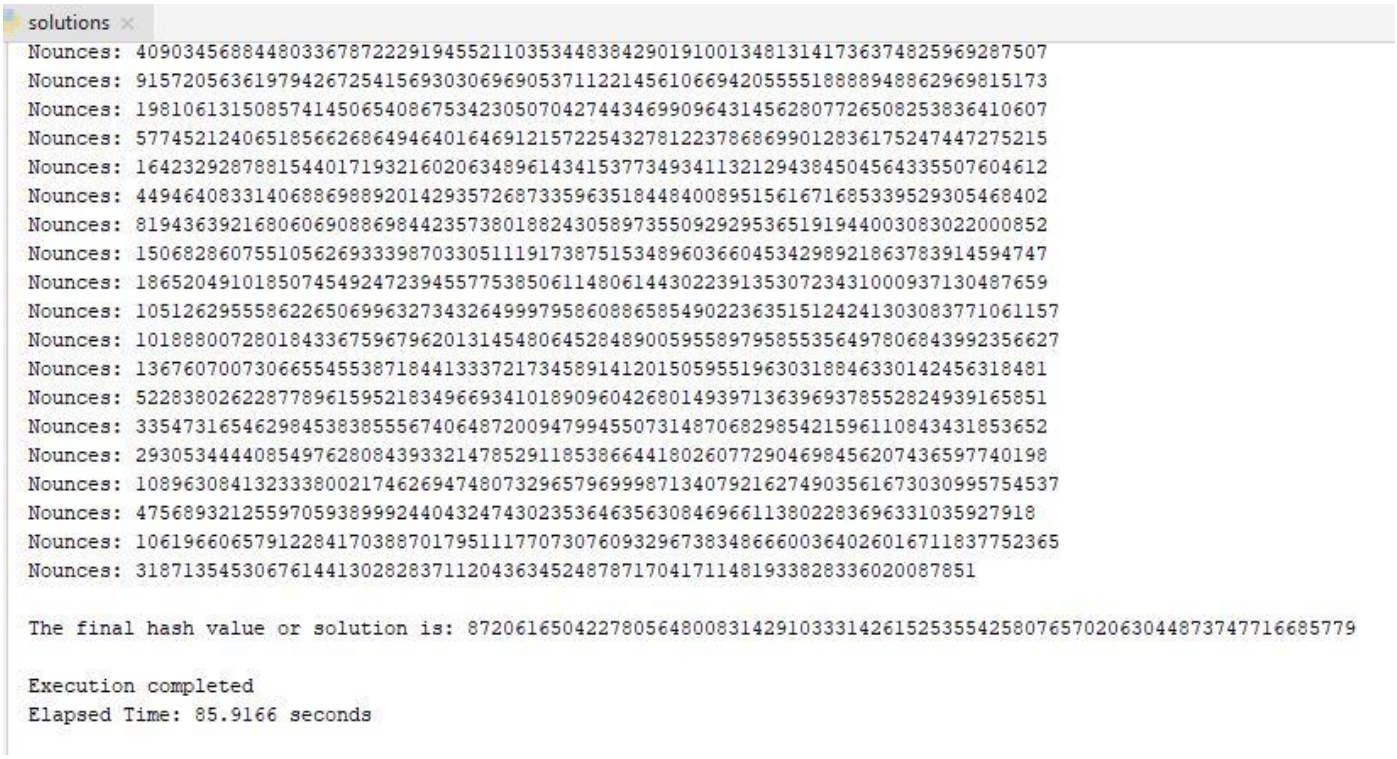
```
import hashlib
max_nonce=1                                    #only 1 verification
def verification(message,solution,target): #inputs message and solution into hash function and target to compare
    for nonce in range(max_nonce):

        hash_result = hashlib.sha256((str(message) + str(solution)).encode('utf-8'))       #hashfunction
        hex_dig = hash_result.hexdigest()
        hash_bin = str(bin(int(hex_dig,16)))
        hash_bin=hash_bin[2:]
        hash_10 = int(hash_bin,2)
        target_10 = int(target,2)
        print ('solution value is:',hash_10)
        if hash_10 <= target_10:
            print(" ")
            print('The final hash value is verified or solution is 1')      #1 if correct

        else:
            print('The solution is 0 ')                                      #0 if incorrect
```

The output of verification is:



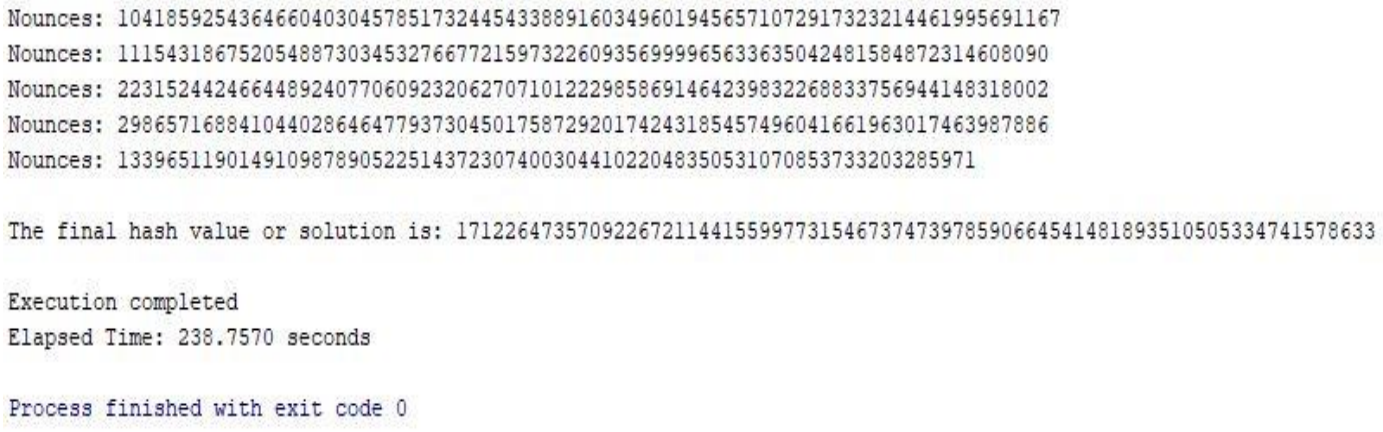
The performance of system for different d values are shown below, d=21,22,23,24,25,26

D=21



Time took =85.9166 seconds.

D=22, Time took=238.7570 seconds.



D=23, Time took = 743.3135 seconds

```
Nounces: 493243285896555200599702272349052270369845721083214443907831741361957205787l
Nounces: 19547342950479288387096794744792165211037138503183553894802375110454638427018
Nounces: 699797758939813977434409415757465551458613974399155074103392593376771585740 25
Nounces: 4448252638347297743823858125496189698592375020634291223854775495685340880879 4
Nounces: 629943434670403571886916241071937825162128495274898676718508697459019818553 44
Nounces: 1127779245781763004252151145302077072242736650856089777745263982162417713524 46
Nounces: 971577944553809083411478482364806179997962329037322287906274511840603124730 33
Nounces: 673215367784383103946626951091546907047651406379388857777984471322301852095 19
Nounces: 1170118422312527679124679240678232424959698864897786185571784452072949 5

The final hash value or solution is: 9811034719960821330118316683055911129631586062953194450024725925759373494866 8

Execution completed
Elapsed Time: 743.3135 seconds

Process finished with exit code 0
```

D=24 , Time took=1235.5470 seconds.

```
Nounces: 4963111690296347231706520329221920214135337817366845803016220136044858750302 6
Nounces: 4972253775094408204726396286471151687178797853294123561327911447634019038578 5
Nounces: 6460598311084237489229808842979214424810520840495124281310983895927878184723 96
Nounces: 1329418778025915015553482434956513698769856594358121445622559740923540604216 6
Nounces: 1802463197855654878783524341590724443925519004264554194473614128028747822866 7
Nounces: 2138771983223232777307708067816111842631627670994972970632763323429599094707 4
Nounces: 5103452796618845314230328080970331971568349328308021695992011855769212271040 9
Nounces: 7125497633409857040431082573541333929613480647304628542783378824545 64044757065
Nounces: 1477355859305254615699397804768519635855124788652858852754374444323749296091
Nounces: 5932768761487587705893073609673661571668550056398496753842434003368395

The final hash value or solution is: 1015966380253112142863758260548285758763381720812047444534806887032016862017 31

Execution completed
Elapsed Time: 1235.5470 seconds

Process finished with exit code 0
```

D=25, Time took =2924.0309 seconds .

```
The final hash value or solution is: 2739655151900256154464783529618718171926375751409678216775255277969881116694 4

Execution completed
Elapsed Time: 2924.0309 seconds

Process finished with exit code 0
```